Miscellaneous Library Features Solutions

Copyright © James Raynard 2021

Pass by reference by value?

- Explain why the code below does not change the value of n
- How could it be made to work as expected?

```
void incr(int& number, int& value) {        // Function with reference argument
   number = value;
   ++value;
}

int n{2};
auto incr_n = bind(incr, _1, n);            // Bind n as second argument of incr
incr_n(number);                             // Call it with bound argument
cout << n << endl;
```

Copyright © James Raynard 2021

Pass by reference by value?

- std::bind() can only capture variables by value
- The returned object, incr_n, will therefore have a copy of the variable n and not a reference to it
- When the object is called, the argument passed to incr() will be a reference to this copy of n, not the original variable
- When incr() modifies this reference argument, the original variable n will be unaffected
- If the argument to bind is wrapped in a call to std::ref(), bind will be passed an object which contains a reference to n
- This reference to n will then be passed to incr(), which will modify the original variable n through the reference

Copyright © James Raynard 2021

fstream constructor

- Write a simple program which passes an std::string object as the argument to the constructor of std::fstream

Copyright © James Raynard 2021

Converting between strings and numbers

- Write a simple program which uses std::to_string() to convert a number to an std::string
- Modify your program to convert this std::string to
  - An int
  - A double

Copyright © James Raynard 2021

Sequential containers data() member function

- What does the data() member function do?
  - It returns the container's internal array
- How could this be useful?
  - For passing std::vector and std::string instances to functions with a C interface
- Write a simple program which demonstrates the use of the data() member function

Copyright © James Raynard 2021